Proactive utilization of proximity-oriented information inside an Agent-based Framework

Vincenzo Loia, Witold Pedrycz*, Sabrina Senatore and **Maria I. Sessa**
Dipartimento Matematica e Informatica - Universita' di Salerno
S. Allende - 84081 Baronissi (SA), Italy

*Department of Electrical and Computer Engineering
University of Alberta, Edmonton, Canada

Abstract

This work presents the formal and practical design of agents skilled to help a user in achieving personalized navigation, by recommending related documents according to the user's sensibility shown in similar-page searching mode. Our agent-based approach is based on the integration of different techniques and methodologies into a unique platform featuring: user profiling, fuzzy multi-sets, proximity-oriented fuzzy clustering, and knowledge based discovery technologies. Each of these approaches serves to solve one facet of the general problem (to carry out personalized web search by discovering documents relevant to the user) and is treated by specialized agents that achieve the final functionality through cooperation and task distribution.

Introduction

The heterogeneity and the lack of standard structures on the Web is becoming the prominent hitch for the navigation and mining activities and even looking for the right items returned by search engines can be a boring and time consuming task. Essentially, search engines are employed to index or categorize the web resources paying attention to relevant words, but it is not always easy to capture the real context of a query just using keywords, which are sometimes confuse, generic and not explicative. A user defines a query for some specific information, but wades through a huge quantity of web items, gathering irrelevant data and often losing the original objective. One way to face these problems is to adopt user profiling techniques, which exploiting the knowledge acquired by past user navigation sessions, infer the right context of the searching/browsing. Within this aim, agents have played an important role in this last decade. (Han *et al.* 1998; Chau *et al.* 2003; Pazzani M. J. and Billsus D. 2002) are just some of the many research experiences reporting how agents explore the Web, categorize the results and then use automatically generated categories to further explore the Web: among these issues, we focus on the automatic categorization of web documents, exploited by the user agents to find new related documents most closely to the starting set.

The work here described represents a further step towards the design of agent-based architectures for advanced web searching/navigation (Loia V., Senatore S. and Sessa M.I. 2003; 2002): the enhancement that we report in this work consists in the design of a proactive agent-oriented facility useful to better define and refine the user profile, combining different granularity-based matching of web documents (such as structure, layout, content, etc).

Outline of the Architecture

Our assistance services framework aims to provide suggestions about similar web documents: according to the user's interests and thanks to the knowledge acquired during the navigation, the system provides confidential hints on documents that are similar to other resources seen by the user. Our architecture is composed of a set of agencies (agent-based components) depicted in Figure 1.

Logger Agency The user's activity is observed and saved during the navigation in order to build updated user profiles, by analyzing some factors: the spent time on visited web page, frequency by which the user returns on the same page, the bookmarks, keyword lists, links, and so on.

Context Agency User profiles tend to represent interest of users over a long-term and typically focus on the topic of the query; our approach is intended to not referring only to the topic but rather addressing the context of user's interest. This is the job of the Context Agency, whose details are given in the following.

Proximity Hinter Agency The user interacts with the system providing a personal evaluation on a set of web pages proposed by the system. These values are passed to system in a digest form for the clustering module.

Clustering Agency It provides the classification of given web pages. We use an extension of fuzzy clustering, called proximity-based fuzzy clustering (Loia V., Pedrycz W., Senatore S. 2003a; 2003b). It allows users to influence the basic classification through the introduction of some criteria about how two web pages are similar/proximal or related (according to the topic, the layout, the user's evaluation). The basic algorithm takes in input the selected web pages as weighted vectors with associated proximity hints.

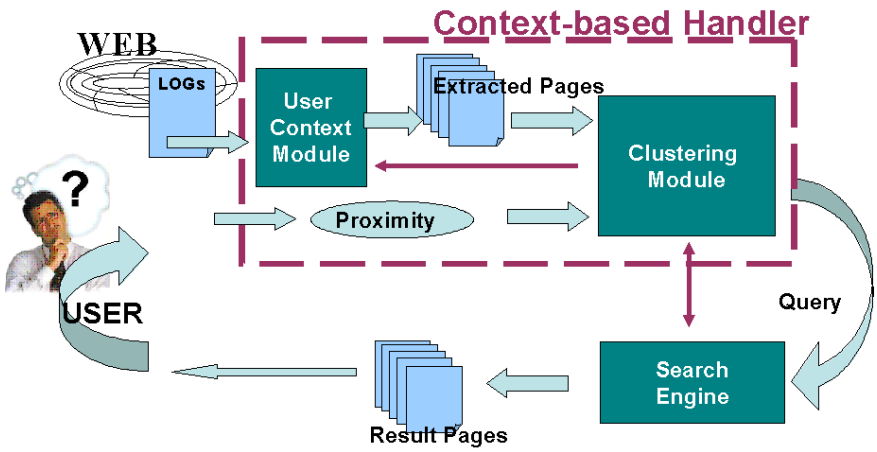

Figure 1: Overall Architecture

Search Engine Agency This layer carries out the effective searching using popular Web search engines. At this moment, the systems works with Google, using Soap (Simple Object Access Protocol) as communication bridge with the other agencies.

Working flow

The system maintains tracks of the user's navigation activities through the web session logs, in order to build updated custom profiles. The obtained results are then sent to components specialized to treat the contextuality of the searching (see Figure 1). The context area analyzes the web pages, by extracting, after a filtering operation, all the relevant information to forward to the clustering agency. At the reception of this information, the clustering agency elaborates the received digest-formed data and applies the proximity-based clustering to classify the visited web pages and to extract the relevant subjects or topics.

At this point the starting training phase is terminated; the extracted topics are employed in new search activities to find new relevant documents. Figure 2 gives a snapshot of a typical session: during the browsing, the agent interface may blink on the screen. If the user clicks on the button *Go to hints* a new panel appears: some urls are given (similar pages), with relative relevance degrees (in the range 0-100) that can be used to (re)formulate the personal judgment. The Proximity Hinter Agency saves the user's feedback for the next iteration, that will be used in the clustering phase. The overall process is then iterated more times, in order to converge towards the satisfaction of the user's requests that better characterize the personal profile.

Context Agency

This module works accomplishing the following activities concurrently:

1. analyzes the log files in order to extract relevant information to define or update user profiles;
2. recovers the web pages referenced in the logs in order to extract topics, relevant keywords, and transform the data in an appropriate format to send to the clustering module.
3. evaluates discovered web documents both considering the user judgments about the proximity or dissimilarity between them or proactively generates some relation between them through the analysis of different views of a web page.

Each web page referenced into the logs session is parsed to build-up a vector space representation of the page. Due to the absence of a standard in the construction of a Web document, the interpretation of a web page is given to a set of goal-oriented extractor agents (Loia V., Senatore S. and Sessa M.I. 2002) that are able to discern the level of analysis of a web page, extract the information enclosed in the tags and finally transform it into a digest form useful for defining the feature space of clustering process. More details on these wrappers are given in the following.

Wrappers

The context analysis is managed by agent-based wrappers whose goal is to extrapolate useful knowledge from a web page and elaborate it. Some wrappers concern the level-based extraction, other operate to identify proactively similarity in the structure of two or more web pages, when no user opinions are provided. More in detail, we distinguish:

- master-extractor: this agent realizes a preliminary analysis of the Web page operating like an indexing: it filters all phrases in the document, removing stop-words, stemming the terms, part-of-speech tagging, etc. In this way, it captures more recurrent words, evaluating the occurrence frequency (TDF-IDF method) in the page and in the session logs. Terms with highest occurrence contribute to describe the user profile, although they are candidate to represent the feature space.
- extractor-layout: this agent examines the framework of a Web page, considering some structural part, relative

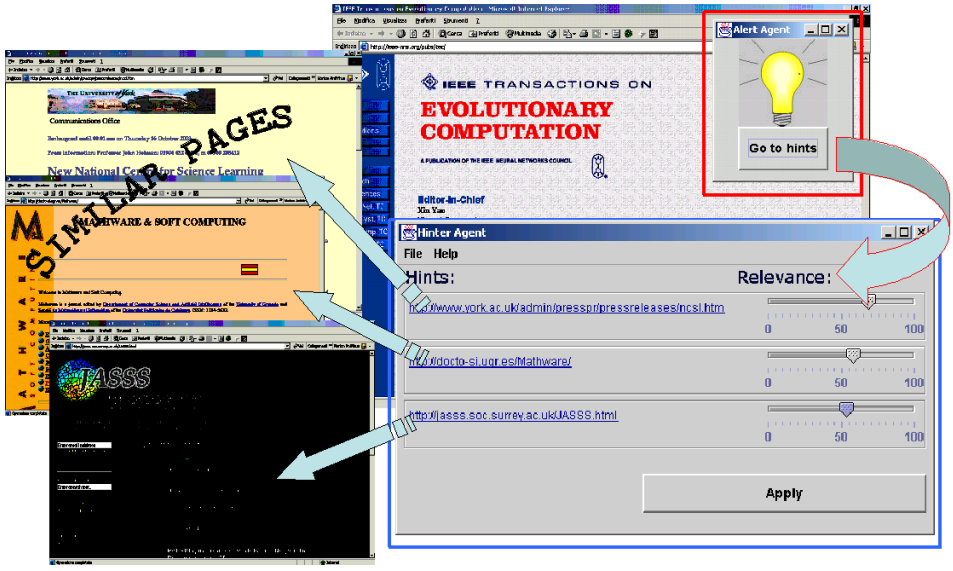

Figure 2: Agent Interface's interaction during the user's navigation

to the layout of the page (frameset, table, etc.). Furthermore, it realizes a comparison (skeleton level) between two Web documents.

- extractor-context: its ability is to process specific sections of a Web page (HTML tags as <title>, <p>,<h1>, <meta>, <a href>, etc.), to extract textual patterns that, according to some specific rules, enables to build a fuzzy multiset to represent the membership of terms in the tags.

- extractor-correlation: this agent works with the extractor-context agent, in order to assign the similarity values defined beetwen the words of the feature space and the terms extracted from the context of web page through a relation that express how two terms are semantically close.

Cooperation Activitites

Each agent provides own contribution to increase the knowledge relative to the web page; in particular for each filtered term t_i, enclosed in a specific tag (provided by the extractor-context agent), correlations with the selected topics (described by terms $w_1, w_2, \ldots, w_n$ in feature space) are computed by the extractor-correlation agent exploiting similarity values $R(w_j, t_i)$. These values are employed to define the fuzzy multiset relative to a selected web page. Possible similarity values, described by a priori fixed semantic-based relation R, characterize a degree of closeness of a term t_i appearing in a tag context with a word of the topic, for instance, w_j:

- If the term $t_i = w_j$ in the current tag then $R(w_j, t_i) = 1$.
- If t_i is a synonym of the word w_j in the current tag, then $R(w_j, t_i) = 0.8$
- If t_i is correlated to the word w_j in the current tag, then $R(w_j, t_i) = 0.6$
- ...
- If t_i has no relation with the term w_j (no correlation is found in the current tag section), $R(w_j, t_i) = 0$.

So for each tag, a list of similarity degrees is constructed in correspondence of encountered terms and with respect to any word w_j in the topic. These sequences of similarity values are then further filtered (through a maximum function's evaluation described in the next section) to elicit the correlation values that better represent the page fuzzy multiset. In fact, each web page is associated with a fuzzy multiset (Miyamoto S. 2003) so to build the document-term matrix for the clustering phase. Another interesting role is played by the extractor-layout agent skilled to compare the web page to others with similar layouts; when the user expressed some (proximity) hints, the pages for which exists the proximity are transformed to be processed by the clustering module, otherwise, interrogating the extractor agents, it is proactively able to deduct structural information to contribute the classification. In order to clarify the approach, it is necessary to give formal aspects in the next section.

Extended Multiset-valued term-document matrix

In order to show how the term-document matrix is built, some formal details are explained. Let T=$\{\tau_1, \tau_2, \ldots, \tau_m\}$ be the HTML tags set. For each page $p_i \in P = \{p_1, p_2, \ldots, p_k\}$, a fuzzy multiset will be constructed, where each element is a word $w_j \in W = \{w_1, w_2, \ldots, w_n\}$ associated with a fuzzy membership computed with respect to specific tags in the web page. In the sequel the symbol $(\alpha_1, \alpha_2, \ldots, \alpha_k)/r_j$ denotes that $(\alpha_1, \alpha_2, \ldots, \alpha_k)$ provides the membership evaluation of the element r_j in a fuzzy multiset.

Definition 1 *$\mathcal{S} : \mathcal{P} \rightarrow \mathcal{F}\,\mathcal{M}(\mathcal{W})$ a function where $\mathcal{F}\,\mathcal{M}(\mathcal{W})$ is a collection of all words in W such that $\forall$ p_i, with $i = 1, \ldots, k$, $\mathcal{S}(p_i)$ is the fuzzy-multiset on W defined as:*

$\mathcal{S}(p_i)$= {($\mu_{\tau 1}(w_1)$, $\mu_{\tau 2}(w_1)$, ... , $\mu_{\tau m}(w_1)$)/w_1), ($\mu_{\tau 1}(w_2)$, $\mu_{\tau 2}(w_2)$, ... , $\mu_{\tau m}(w_2)$)/w_2), ... , ($\mu_{\tau 1}(w_n)$, $\mu_{\tau 2}(w_n)$, ... , $\mu_{\tau m}(w_n)$)/w_n) } where τ_h is a tag in T, with $h = 1, \ldots, m$ and $\mu_{\tau_h}(w_j) \in [0, 1]$ is the membership value of term w_j with $j = 1, \ldots, n$ in the tag τ_h.

Thus, each row of the term-document matrix is a vector, related to a web page, that represents a fuzzy multiset of words in W.
In order to characterize the membership value $\mu_{\tau_h}(\mathrm{w}_j)$, the following definition is given:

Definition 2 *Given a tag τ_h in a web page p, its context $C_{\tau h}$ is given by a set of terms: $C_{\tau h}$= {t_1, t_2, ... t_n} extracted from τ_h through automatic indexing technique. We define:*

$\mu_{\tau h}(w_j)$= $\max_{t_i \in C_{\tau h}}(R(w_j, t_i))$ $\forall \tau_h \in T$,
where $w_j \in \mathcal{W}$ and $R(w_j, t_i)$ is a priori defined similarity value.

In particular, the context C_p of a web page p is given as:

$$\mathrm{C}_p = \cup_{\tau h}\, \mathrm{C}_{\tau h},\ \forall \tau_h \in \mathrm{T}$$

This membership value represents the best similarity value calculated by extractor agents in each tag with respect to word w_j of feature space. The Figure 3 shows a parsed web page where terms enclosed in selected tags (in the figure we only focus on *TITLE*, *Meta*, *H3* and *A HREF* tags) are extracted and, through extractor agents' action, the semantic correlations with a given feature word (in the example of Figure 3, *security*) are expressed through a correspondent sequence of membership values obtained as the maximum similarity value of terms in the tag context. In this way, extending this approach to all words of feature space, for each web page, a correspondent fuzzy multiset is built as a row entry in term-document matrix.

Proximity based Fuzzy clustering

Proximity-based FCM, or shortly P-FCM (Loia V., Pedrycz W., Senatore S. 2003a), is an extension of the well-known fuzzy c-means clustering (FCM) algorithm (Bezdek, J.C. 1981) particularly useful for Web exploration and data organization on the Web. This approach can offer a relatively simple way of improving the Web page classification according with the user interaction with the search engine.

In fact, many factors may play an important role in a human judgment concerning the "proximity" of Web pages (layouts, backgrounds, links, texts, ...). and are difficult to quantify and to translate into computationally meaningful features. Usually the textual data is the most evident and it is almost the exclusive contributor to the feature space when determining structures in a collection of Web pages. The use of the proximity hints can compensate for the consideration of a subset of the feature space by capturing hypermedia or cognitive information.

An overview on P-FCM algorithm

The concept of proximity between two objects (patterns) is one of the fundamental notions of high practical relevance. Formally, given two patterns "a" and "b" , their proximity, prox(a, b), is a mapping to the unit interval such that it satisfies the following two conditions

prox (a, b) = prox (b, a) symmetry
prox (a, a) =1 reflexivity

Table 1: An optimization flow of the P-FCM algorithm

Given: specify number of clusters, fuzzification coefficient, distance function and initiate a partition matrix (generally it is started from a collection of random entries), termination condition (small positive constant ϵ).

Repeat — main external loop
  Compute prototypes and partition matrix using standard expressions encountered in the FCM method
  Repeat — internal optimization loop
    Minimize some performance index V guided by the collection of the proximity constraints
  Until no significant changes in its values over successive iterations have been reported (this is quantified by another threshold δ)
Until a termination condition has been met (namely, a distance between two successive partition matrices does not exceed ϵ).

The notion of proximity verifies a minimal set of requirements; what we impose is straightforward: "a" exhibits the highest proximity to itself and the proximity relation is symmetric. In this sense, we can envision that in any experimental setting, these two properties can be easily realized. P-FCM computing scheme comprises of two nested phases, as given in Table 1. The upper level deals with the standard FCM computing (iterations) and follows the well known scheme encountered in the literature, while the one nested is aimed at the accommodation of the proximity requirements and optimizes the partition matrix on this basis. The accommodation of the proximity requirements (constraints or hints) is realized in the form of a certain performance index whose minimization leads us to the optimal partition matrix. As stated in the problem formulation, we are provided with pairs of patterns and their associated level of proximity. The partition matrix U (more specifically the induced values of the proximity) should adhere to the given levels of proximity. Bearing this in mind, the performance is formulated as the following sum.

$$V = \sum_{k_1=1}^{N} \sum_{k_2=1}^{N} (\hat{p}[k_1, k_2] - p[k_1, k_2])^2 b[k_1, k_2] d[k_1, k_2] \tag{1}$$

The notation $\hat{p}[k_1, k_2]$ is used to describe the proximity level induced by the partition matrix. The value $d[k_1, k_2]$ denotes the distance between the two corresponding patterns while $p[k1, k2]$ is the proximity level provided by the user or data analyst; $b[k_1, k_2]$ assumes binary value: it returns 1 if there is a proximity hint for this specific pair of the patterns, that is k_1 and k_2, otherwise the value of $b[k_1, k_2]$ is set up to zero (meaning that there is no proximity hint for the specific pair of data). Major details about the P-FCM algorithm and its formulation are given in (Loia V., Pedrycz W., Senatore S. 2003a).

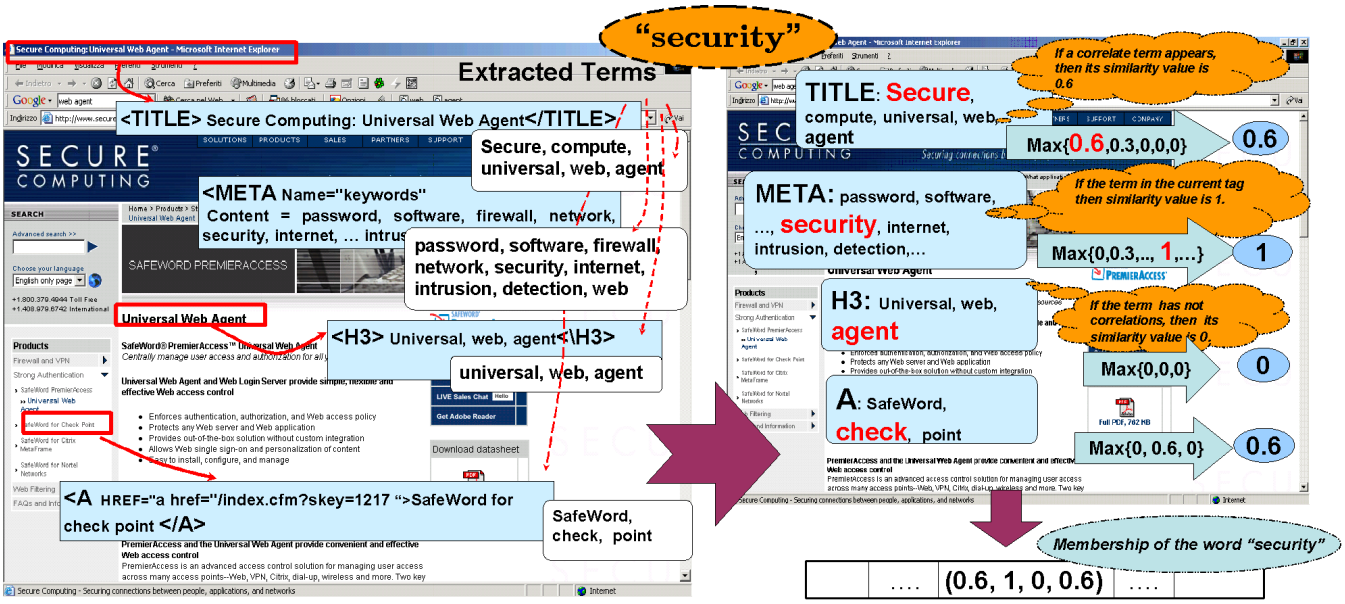

Figure 3: Construction of the membership value of the term "security" in the fuzzy multiset entry

Experimental Results

Table 2 shows the initial training phase. For each user (10 in total), the system maintains track of the navigation activities: the number of web access sessions, the visited web pages, the keywords selected in the wrappers-step for the clustering and the effective words that the system propone to web searching activity. The last column of the table shows the user judgment on the relevance of pages returned by the searching, during the navigation. Table 3 shows results obtained after a training phase: each new iteration (in total three) identifies the process of reclassification using user feedback. As evidenced by our first experimentation, the *precision* of the searching improves until the system becomes stable, soon after the third iteraction.

Conclusions

Search engines help users speeding up the information discovering activity, but often the unclear context of the searched topic described by ambiguous keywords with multiple meanings, makes difficult to center real arguments and, consequently to retrieve the meaningful web documents. In order to provide a robust approach to treat the difference between the information extracted from the net and the information useful to the user within a certain topic, we have embedded into a previous system an additional functionality for user assistance: recommending related documents according to the user' feedback shown in similar-page searching mode.

References

Bezdek, J.C. 1981. *Pattern Recognition and Fuzzy Objective Function Algorithms*. N. York: Plenum Press.

Chau, M.; Zeng, D.; Chen, H.; Huang, M.; and Hendriawan, D. 2003. Design and evaluation of a multi-agent collaborative web mining system. *Decision Support Systems* 35(1):167–183.

Han, E.; Boley, D.; Gini, M.; Gross, R.; Hastings, K.; Karypis, G.; Kumar, V.; Mobasher, B.; and Moore, J. 1998. WebACE: A web agent for document categorization and exploration. In Sycara, K. P., and Wooldridge, M., eds., *Proceedings of the 2nd International Conference on Autonomous Agents*, 408–415. New York: ACM Press.

Loia V., Pedrycz W., Senatore S. 2003a. P-FCM: A Proximity-Based Fuzzy Clustering. *accepted on Special Issue of Fuzzy Sets and Systems on Web Mining using Soft Computing*.

Loia V., Pedrycz W., Senatore S. 2003b. P-FCM: A Proximity-Based Fuzzy Clustering for user-centered Web applications. *International Journal of Approximate Reasoning (IJAR)* 34(23):121–144.

Loia V., Senatore S. and Sessa M.I. 2002. Discovering Related Web Pages through Fuzzy-Context Reasoning. In *Proceedings of 11th IEEE International Conference on Fuzzy Systems*, volume 1, 150 –155. Hawaii: IEEE PRESS.

Loia V., Senatore S. and Sessa M.I. 2003. Similarity-based SLD Resolution its Role for Web Knowledge Discovery. In *to appear the Special Issue on "Advances in Possibilistic Logic and Related Issues"*. Fuzzy Sets & Systems.

Miyamoto S. 2003. Information clustering based on fuzzy multisets. *Information Processing and Management* 39 2:195–213.

Pazzani M. J. and Billsus D. 2002. Adaptive Web Site Agents. *Autonomous Agents and Multi-Agent Systems* 5:205–218.

Table 2: Experimental results: preliminary setup

User N.	*User sessions*	*# visited pages*	*keywords*	*hints*	% precision*
# 1	41	92	learn, teach, computer, knowledge, tutor, howto, retail, business, sell, product, distance, education, guide, online	learn, teach, online, product, knowledge, business, education	14 %
# 2	38	78	music, guitar, concert, play, drums, soccer, team, campionato, partita, torneo, jazz, entertainment	music, soccer, partita, play, concert	16 %
# 3	27	56	transfer, gift, photo*, image, software, filter, digital, manipulat*, concert, music, journal, promot*, portrait	photo, software, image, music, concert, gift	19 %
# 4	26	67	web, www, internet, net, network, software, search, crawler, bot, spider, robot, browser, conference	internet, browser, conference, crawler, bot	20 %
# 5	36	79	develop*, compan*, manag*, resource, business, finance, employ*, web, investiment, software, education	business, finance, employ*, investiment, develop*	27 %
# 6	19	83	soft computing, genetic algorithm, fuzzy logic, fuzzy systems, fuzzy information, fuzzy sets, neural networks, expert system, artificial intelligence, information retrieval, data mining	soft computing, fuzzy logic, neural networks, artificial intelligence, fuzzy systems	31%
# 7	16	27	wallpaper, ram, screeensaver*, image*, commerce, icon*, vaio, clipart*, graphic*, quality, intel, processor, hardware, free, computer, desktop	ram, screeensaver*, image*, vaio, clipart*, graphic*, computer, intel, processor	27%
# 8	55	78	corso, lezione, lab, laurea, prova, appello, calendario, guida, studente, informatica, diploma, obiettivi, web, scienze, ordinamento	informatica, studente, laurea, guida, appello, corso	46%
# 9	18	59	easilix, software, operat*, system, freeware, license, download, linux, install*, utilit*, graphic*, free, distribution, redhat, suse	download, linux, install*, operat*, system, free, redhat, suse	51%
# 10	8	45	digital, camera, product, card, accessor*, canon, kodak, batter*, memory, macchina, fotografica, fotografia, nikon, prezzo, photography, olympus	digital, camera, batter*, memory, card, accessor*	33%

* percentage of pages returned by search engine, selected by the user as relevant

Table 3: Experimental results: after 3 iterations (with users' feedback)

User N.	*Whole User sessions*	*# visited pages*	%precision*		
			I iteration	II iteration	III iteration
# 1	86	140	25%	32%	37%
# 2	55	132	17%	37%	43%
# 3	43	66	22%	24%	23%
# 4	39	96	22%	27%	35%
# 5	41	96	29%	33%	38%
# 6	44	92	33%	38%	39%
# 7	32	88	30%	30%	34%
# 8	115	101	49%	51%	51%
# 9	37	84	66%	73%	73%
# 10	23	142	40%	56%	57%

* percentage of pages returned by search engine, selected by the user as relevant